

CCS Components: State Storage Deep Dive

CCS Components: State Storage Deep Dive Agenda

- Timing
 - 15-20 min: Ulf Intro presentation on CSS Components: State Storage.
 - 40 min: reaction and building future plan
- Technical API discussion
 - Design Issues in Ulf's presentation
 - Review current State Storage component API
 - Anything to fix or extend?
 - Categorise in wider architecture:
 - In VSS Terminology State Storage is 'simple'. Is this component strong coupled (fit) for that use case and something else is needed for 'more powerful' or do we reuse/extend?
 - Simple vs powerful may have multiple dimensions: throughput, features (e.g. events), etc.
 - Related areas (likely will need their own Deep Dive sessions as too big a topic to conclude here)
 - Abstraction APIs
 - To Feeder, to Server
 - Requirements
 - Speed
 - Low vs medium vs high frequency
 - Native for high frequency?
 - Kuksa.val is also a target
 - Last value vs time series
 - VISS protocol has some TS query
 - 'History filtering'
 - Simple support for this in current WAI. Other process (e.g. comms) must tell WAI to start/stop recording. Then VISS client (e.g. cloud) must detect vehicle became disconnected and query for missing data.
 - Kuksa currently has no TS
- Date
 - Monday 23rd May 4pm CET.
- Rough meeting notes:
 - Attendees: Ted Guild, Ulf, Paul, James Murphy, Florian Pinzel, Stephen, Jose Gomez
 - Presentation: [COVESA AMM 2022 - State storage \[PUB\].pdf](#)
 - Origin of component in CCS Project
 - Data structure
 - Actuator model encapsulated (abstracted) in State Storage component
 - VSS Path list
 - SQLite has manager to help create path list.
 - Redis has no need for path list.
 - Design Issues
 - Polling vs event notification
 - Ulf: looked into SQLite event framework but it didn't seem to fit the needs here. Not looked into Redis yet.
 - Discussion of requirements of slow to medium frequency data vs high frequency
 - Futures
 - Timeseries, event notification, app framework etc.